Converting Informal Meta-data to VDM-SL: A Reverse Calculation Approach

FM'99 — Workshop *VDM in Practice!*
20 –21 September
Toulouse
France

F.L. Neves, J.C. Silva & J.N. Oliveira
Departamento de Informática
Universidade do Minho
Braga
Portugal

email: jno@di.uminho.pt
http://www.di.uminho.pt/~jno

Context

Formal Methods

- Industrial application of **FM**s
- Program **calculi**

Industrial Framework

- Legacy systems
- Large-scale **reverse specification**
- VLDBs and **data intensive** applications
- **Data Quality**

R&D Contract KARMA

(Grant P060-P31B-09/97)

= 3 software houses

Sidereus Ltd, Novabase Porto & Novabase Lisbon

\+ **FM Group** at Minho

The Problem

- Standard information system (**IS**) technology is too **informal**

- Hard to **maintain** the quality of fast-growing data in loosely structured IS.
- **"Dirty" data** infection vs **data quality**
- (Very) Expensive *data cleansing / reverse specification* contracts:
  - Telecommunication industry
  - Healthcare
  - Governmental Depts. (*e.g.* Social Security)
  - *etc.*

The KARMA Approach

- Data-quality from a **formal-method** viewpoint
- Formal models for **meta-data** representation
- **Reverse engineering** based on the mathematics of program calculation

Background

- Formal specification and rapid prototyping
- Past experience gathered in reverse specifying U. Minho's **Student Record** database
- Data refinement calculi

Reverse Specification

The usual diagram:

$$\begin{tikzcd} A \arrow[r, "\sigma"] & A \\ C \arrow[u, "I"] \arrow[r, "\tau"'] & C \arrow[u, "I"'] \end{tikzcd}$$

where

- A — abstract state
- C — concrete state
- Code τ implements the behaviour of spec σ
- I is the coupling invariant linking the abstract and concrete states.

One is given the code (τ) and is asked to produce the spec (σ).

What about I? How do we guess this?

Typical Data Quality Context

- Very large IS
- No access to the code (only stored data are available)
- I is at target — one wants to check (hundreds of) business rules across this abstract-to-concrete link.

Functional Data Refinement

Quotation from C. Morgan's book (chapter 17, pp. 173–174):

> *The laws of Section 17.4 dealt with a very general case of data refinement, in which the coupling invariant linking the abstract and concrete states could be anything whatever.* [...]
>
> *This is actually a fairly rare occurrence in everyday program development however: it is much more common for the coupling invariant to be functional from concrete to abstract.* [...]
>
> *The general form for such coupling invariants, called* functional abstractions *is*
>
> $$a = \mathsf{af}\ c \wedge \mathsf{dti}\ c \quad , \qquad \textit{(17.1)}$$
>
> *where af is a function, called the* abstraction function *and dti is a predicate, in which a does not appear, called the* data-type invariant. [...]

"Epic" Data Refinement

We don't want data representation pitfalls:

$\boxed{af \text{ should be surjective ("epic")}}$

Abstraction Invariant Calculus

Inequation $A \leq C$ (*"data structure C implements, or refines data structure A"*) means

- there is an *abstraction function* $A \xleftarrow{af} C$ which has a canonical right-inverse $A \xrightarrow{r} C$ (the *representation function*):

$$af \cdot r = id_A$$

Let ci denote the *concrete invariant* which follows:

$ci\ c \Leftrightarrow$ there exists at least one $a \in A$ such that $c = r\ a$

Then

$$A \cong C_{ci}$$

where C_{ci} denotes the subset of C which satisfies ci.

Reverse Spec via "concrete invariant discharge"

Safe replacement of C-structures by A-structures provided ci is known to hold.

In the IS context ci may correspond to

- structural **integrity constraints**
- ad hoc **business rules**

Can invariant discharge be fully automated?

Not straightforwardly:

There may exist two or more candidate laws for invariant discharge, for instance $B \cong C_\psi$ competing with $A \cong C_{ci}$.

Reverse C either to

A — in case ci holds

B — in case ψ holds

Examples

Concrete state CST consisting of two relational tables sharing the same primary-key:

```
CST :: T1: map A to B
       T2: map A to C;
```

may either be isomorphic to

```
AST1 = map A to BandC;
BandC :: A1: B
         A2: C;
```

in case

```
inv mk_CST(t1,t2) == dom t1 = dom t2
```

holds, or to

```
AST2 = map A to BorC;
BorC = CaseB | CaseC;
CaseB :: K: B;
CaseC :: K: C;
```

in case

```
inv mk_CST(t1,t2) == (dom t1) inter (dom t2) = {}
```

holds.

Inequation

$$A \rightharpoonup B \times C \underset{join}{\overset{unjoin}{\leq}} (A \rightharpoonup B) \times (A \rightharpoonup C) \tag{1}$$

supports the reversal of `CST` into `AST1`.

Inequation

$$A \rightharpoonup (B + C) \underset{cojoin}{\overset{uncojoin}{\leq}} (A \rightharpoonup B) \times (A \rightharpoonup C) \qquad (2)$$

supports the reversal of `CST` into `AST2`.

Relational database schema reverse-specification

Relatively modest set of data-transformation laws, *e.g.*

- Two tables sharing a common (sub)key can be merged:

$$A \rightharpoonup (B \times (C \rightharpoonup D)) \underset{njoin}{\overset{unnjoin}{\leq}} (A \rightharpoonup B) \times (A \times C \rightharpoonup D) \qquad (3)$$

- finite sets "are" special cases of finite mappings:

$$\mathcal{P}\, A \underset{\textit{dom}}{\overset{set2fm}{\cong}} A \rightharpoonup 1 \qquad (4)$$

  cf. `set of A` and `map A to nil`.

  (NB: "intermediate" datatype `nil` is non-standard in VDM-SL)

Development of the KARMA toolset

Two formal specs have been produced and prototyped so far:

Karma

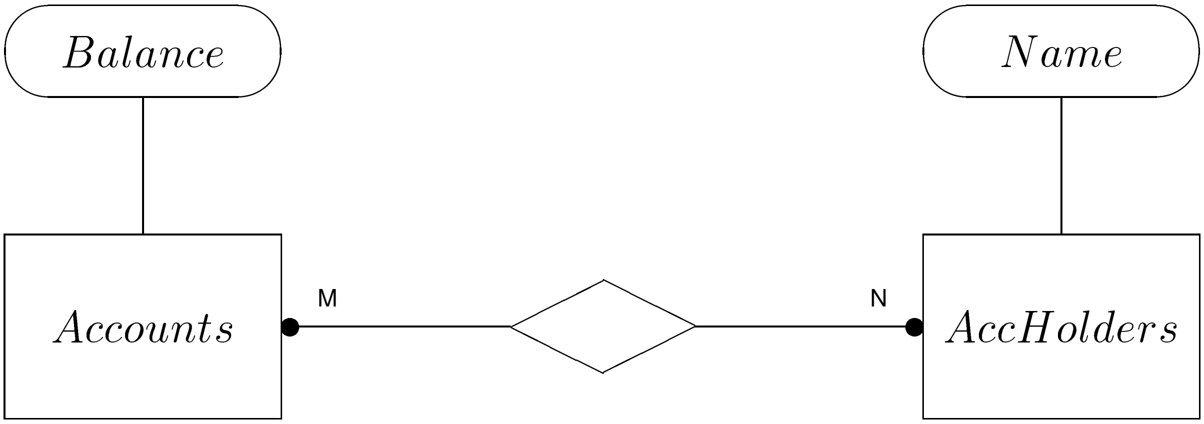
Formal definition of a relational meta-data standard.

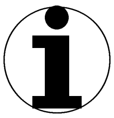
KarmaSets

The ultimate goal of this tool is to deliver concise formal descriptions — written in the ISO standard formal notation VDM-SL — out of poorly structured meta-data.

KarmaSets at work

Toy example — a naïve “bank account management system”:





At any stage the tool can be asked to output the VDM-SL specification of the current version of the model under reversal.

```
-----------------------------------------------------
-- VDM-SL type & data of Sets Abstract model.
-----------------------------------------------------
-- Generated Automatically by KarmaSets tool.
-----------------------------------------------------

types

Sets ::
      P1: set of Inf_1
      P2: set of Inf_2
      P3: set of Inf_3 ;

Inf_1 ::
      P1: AccountId_01
      P2: Balances_meta01 ;

Inf_2 ::
      P1: AccountId_01
      P2: AccHolderId_02 ;

Inf_3 ::
      P1: AccHolderId_02
      P2: Details_meta02 ;

Balances_meta01 ::
      P1: Balance;

Details_meta02 ::
      P1: Name;

AccountId_01 = seq of char;
AccHolderId_02 = seq of char;
Balance = seq of char;
Name = seq of char;
```

Taking integrity constraints of the diagram into account, the main body of this converts to

```
Sets ::
      P1: Inf_1
      P2: map AccHolderId_02 to Details_meta02  ;

Inf_1 ::
      P1: map AccountId_01 to Balances_meta01
      P2: map Inf_2 to nil  ;

Inf_2 ::
      P1: AccountId_01
      P2: AccHolderId_02 ;
```

The two occurrences of attribute `AccountId_01` provide an opportunity to trigger law (3) above:

```
Sets ::
      P1: map AccountId_01 to Inf_1
      P2: map AccHolderId_02 to Details_meta02  ;

Inf_1 ::
      P1: Balances_meta01
      P2: map AccHolderId_02 to nil  ;
```

Finally . . .

Law (4) can be applied to the second component of `Inf_1`:

```
Inf_1 ::
      P1: Balances_meta01
      P2: set of AccHolderId_02  ;
```

Application to Legacy IS

- Legacy information systems are normally poorly documented.
- Reverse-specification decisions:
  - informal inspection (interviews with systems users)
  - concrete-invariant exception statistics.
- The tool keeps track of both the “abstraction function so far” and of the associated representation function.
- Attributes which do not participate in the referential integrity of the information system are factored out as “meta” attributes (*e.g.* `Details_meta02`).

Large IS Partitioning

From which relational tables should one start the analysis?

- Transitive closure of the referential-integrity graph makes it possible to “cluster” closely related meta-data.
- Clusters are identified by numbers which become apparent as figures which the tool attaches to the output VDM-SL identifiers (*e.g.* `01` in `AccountId_01`).

Current work

- We would like to move further to reversing the operations (transactions)
- inference of object-oriented data-models (VDM++)
- inference of temporal data models

The Student Records Case Study

More than 50 tables partitioned in three clusters:

- Students (38%)
- Courses (17%)
- Others

VDM-SL specification output by the tool

```
-----------------------------------------------------
-- VDM-SL type & data of Sets Abstract model.
-----------------------------------------------------
-- Generated Automatically by KarmaSets tool.
-----------------------------------------------------

types

Sets ::
      P1: map NUMERO_01 to Inf_1
      P2: map CURSO_02 to Inf_2
      P3: map DISC_09 to Inf_3
      P4: map PLANO_08 to Inf_4
      P5: map OPNAME_16 to Inf_5
      P6: map CP1_06 to CODIGO_POSTAL_meta07
      P7: map FREG1_07 to Inf_6
      P8: map CPC_12 to CPCS_meta09
      P9: map DEPART_14 to Inf_7
      P10: map CODIGO_DISTRITO_10 to DISTRITOS_meta15
      P11: map SUPER_15 to Inf_8
      P12: map ENDER_13 to ENDERUM_meta17
      P13: map COD to Inf_9
```

```
      P14: map COD to Inf_10
      P15: map Inf_11 to EXAMES_TITULO_meta25
      P16: map FREGN_03 to Inf_12
      P17: map LOCDOCID_05 to LOCEMBI_meta31
      P18: map PAISN_04 to PAISES_meta35
      P19: map TITULO_11 to PAUTASTIT_meta38  ;

Inf_1 ::
      P1: Inf_13
      P2: CLASSIFCTL_meta06
      P3: map ANO_LECTIVO to Inf_14
      P4: map DISC_09 to Inf_15
      P5: map DISC_09 to Inf_16
      P6: map Inf_17 to CPTERMOSCLASS_meta10
      P7: map Inf_18 to EQUIV_meta18
      P8: map DISC_09 to Inf_19
      P9: map DATAHOM to Inf_20
      P10: map Inf_21 to Inf_22
      P11: map Inf_23 to INSCRICOES_DEZEMBRO_meta28
      P12: map Inf_24 to Inf_25
      P13: map Inf_26 to INSCRICOES_SETEMBRO_meta30
      P14: map ANO_LECTIVO to Inf_27
      P15: map Inf_28 to PAUTASCLASS_meta36
      P16: map PLANO_08 to PLANOS_COMPLEMENTARES_meta42
      P17: map ANO_LECTIVO to RECIBOS_meta43
      P18: map Inf_29 to TERMOSCLASS_meta44
      P19: map ANO_LECTIVO to Inf_30  ;

Inf_13 ::
      map CURSO_02 to ALSCONC_meta01  |
      Inf_31 ;

Inf_31 ::
      P1: FREGN_03
      P2: PAISN_04
      P3: PAISN_04
      P4: LOCDOCID_05
      P5: CP1_06
      P6: FREG1_07
      P7: CP1_06
      P8: FREG1_07
      P9: CURSO_02
      P10: PLANO_08
      P11: ALUNOS_meta02 ;

Inf_14 ::
      P1: CURSO_02
      P2: PLANO_08
      P3: ALUNOS_INSCRITOS_meta03 ;

Inf_15 ::
      P1: DISC_09
      P2: CLASSIF_meta04 ;

Inf_16 ::
      P1: DISC_09
      P2: CLASSIF0_meta05 ;

Inf_17 ::
      P1: ANOLEC
      P2: CURSO_02
      P3: PAG ;
```

```
Inf_18 ::
      P1: OFICIO
      P2: ANOLEC ;

Inf_19 ::
      P1: DISC_09
      P2: EQUIVALENCIAS_meta20 ;

Inf_20 ::
      P1: SUPER_15
      P2: ESTAGENG_meta23 ;

Inf_21 ::
      P1: ANO_LECTIVO
      P2: DISC_09 ;

Inf_22 ::
      P1: DISC_09
      P2: INSCRICOES_meta27 ;

Inf_23 ::
      P1: ANO_LECTIVO
      P2: DISC_09 ;

Inf_24 ::
      P1: ANO_LECTIVO
      P2: TITULO_11
      P3: DISC_09
      P4: NUM_CHAMADA ;

Inf_25 ::
      P1: DISC_09
      P2: INSCRICOES_EPOCA_meta29 ;

Inf_26 ::
      P1: ANO_LECTIVO
      P2: DISC_09 ;

Inf_27 ::
      P1: CURSO_02
      P2: PLANO_08
      P3: NAO_INSCRITOS_meta32 ;

Inf_28 ::
      P1: ANOLEC
      P2: CURSO_02
      P3: PAG ;

Inf_29 ::
      P1: ANOLEC
      P2: CURSO_02
      P3: PAG ;

Inf_30 ::
      P1: CURSO_02
      P2: PAISN_04
      P3: LOCDOCID_05
      P4: PAISN_04
      P5: FREGNAT
      P6: CP1_06
      P7: FREG1_07
```

```
      P8: CP1_06
      P9: FREG1_07
      P10: OPNAME_16
      P11: PLANO_08
      P12: VER_ALUNOS_meta46 ;

Inf_2 ::
      P1: PLANO_08
      P2: CPC_12
      P3: CURSOS_meta12
      P4: DISC_09
      P5: map Inf_32 to Inf_33
      P6: map IDENT to EQUIV0_meta19
      P7: map Inf_34 to nil
      P8: map Inf_35 to nil
      P9: map ANO_LECTIVO to PAGINAS_CURSO_meta34
      P10: map Inf_36 to Inf_37
      P11: map Inf_38 to Inf_39  ;

Inf_32 ::
      P1: ANOLEC
      P2: PAG ;

Inf_33 ::
      P1: DISC_09
      P2: TITULO_11
      P3: CPTERMOSDISC_meta11 ;

Inf_34 ::
      P1: IDENT
      P2: DISC_09 ;

Inf_35 ::
      P1: IDENT
      P2: DISC_09 ;

Inf_36 ::
      P1: ANOLEC
      P2: PAG ;

Inf_37 ::
      P1: DISC_09
      P2: TITULO_11
      P3: PAUTASDISC_meta37 ;

Inf_38 ::
      P1: ANOLEC
      P2: PAG ;

Inf_39 ::
      P1: DISC_09
      P2: TITULO_11
      P3: OPNAME_16
      P4: TERMOSDISC_meta45 ;

Inf_3 ::
      P1: DEPART_14
      P2: DISCS_meta14
      P3: map ANO_LECTIVO to nil
      P4: map Inf_40 to Inf_41
      P5: map DISC_09 to nil
      P6: map Inf_42 to Inf_43  ;
```

```
Inf_40 ::
      P1: ANO_LECTIVO
      P2: EPOCA
      P3: CHAMADA ;

Inf_41 ::
      P1: CURSO_02
      P2: EXAMES_meta24 ;

Inf_42 ::
      P1: ANO_LECTIVO
      P2: COD_FUNCAO ;

Inf_43 ::
      P1: CURSO_02
      P2: SUPER_15 ;

Inf_4 ::
      P1: PLANOS_meta41
      P2: map Inf_44 to Inf_45
      P3: map Inf_46 to nil  ;

Inf_44 ::
      P1: ANT_17
      P2: ANO ;

Inf_45 ::
      P1: DISC_09
      P2: PLANDISC_meta40 ;

Inf_46 ::
      P1: ANT_17
      P2: ANT_17 ;

Inf_5 ::
      P1: OPERADORES_meta33
      P2: map PROGRAMA to PERMISSOES_meta39  ;

Inf_6 ::
      P1: CODIGO_DISTRITO_10
      P2: CONCELHOS_meta08 ;

Inf_7 ::
      P1: ENDER_13
      P2: DEPARTS_meta13 ;

Inf_8 ::
      P1: DEPART_14
      P2: ENDER_13
      P3: DOCS_meta16 ;

Inf_9 ::
      P1: ENDER_13
      P2: ESCOLAS_meta21 ;

Inf_10 ::
      P1: CP1_06
      P2: ESTAB_meta22 ;

Inf_11 ::
      P1: ANO_LECTIVO
```

```
      P2: EPOCA
      P3: CHAMADA ;

Inf_12 ::
      P1: FREG1_07
      P2: FREGUESIAS_meta26 ;

CODIGO_POSTAL_meta07 ::
      P1: LOCALIDADE;

CPCS_meta09 ::
      P1: TRAT
      P2: NOME
      P3: END1
      P4: END2
      P5: END3;

DISTRITOS_meta15 ::
      P1: NOME;

ENDERUM_meta17 ::
      P1: END1
      P2: END2
      P3: END3;

EXAMES_TITULO_meta25 ::
      P1: DESCRICAO;

LOCEMBI_meta31 ::
      P1: LOCAL;

PAISES_meta35 ::
      P1: NOME;

PAUTASTIT_meta38 ::
      P1: NOME;

CLASSIFCTL_meta06 ::
      P1: ESTADO
      P2: QUANTOS
      P3: OPNAME
      P4: OPDATE;

CPTERMOSCLASS_meta10 ::
      P1: REGIME
      P2: CLASS
      P3: NUM_ORD;

EQUIV_meta18 ::
      P1: TIPOOF
      P2: QUEM
      P3: DATAENV
      P4: DATAREC
      P5: DATAHOM
      P6: DATAPED;

INSCRICOES_DEZEMBRO_meta28 ::
      P1: TIPO
      P2: OPNAME
      P3: OPDATE;

INSCRICOES_SETEMBRO_meta30 ::
```

```
      P1: TIPO
      P2: OPNAME
      P3: OPDATE;

PAUTASCLASS_meta36 ::
      P1: REGIME
      P2: SOBRE
      P3: NUM_ORD;

PLANOS_COMPLEMENTARES_meta42 ::
      P1: MEDIA_PARCIAL
      P2: FACTOR_PONDERACAO;

RECIBOS_meta43 ::
      P1: MATRICULA
      P2: INSCRICAO
      P3: SEGURO
      P4: CARTAO
      P5: GUIA
      P6: COTA
      P7: TOTAL
      P8: OPNAME
      P9: OPDATE;

TERMOSCLASS_meta44 ::
      P1: REGIME
      P2: CLASS
      P3: NUM_ORD;

ALSCONC_meta01 ::
      P1: CLASSFINAL
      P2: CLASSESTAGIO
      P3: DATCONC
      P4: DATCARTA
      P5: RECCARTA
      P6: PROVISORIA;

ALUNOS_meta02 ::
      P1: NOME
      P2: NOMEORD
      P3: SEXO
      P4: FALECIDO
      P5: DATN
      P6: E_PALOP
      P7: DOCID
      P8: NMPAI
      P9: NMMAE
      P10: RUA1
      P11: LOC1
      P12: TEL1
      P13: RES1RES2
      P14: RUA2
      P15: LOC2
      P16: TEL2
      P17: DATM
      P18: ANO
      P19: COD_RAMO
      P20: COD_ESPECIALIZACAO
      P21: DATVACINA
      P22: ACTUALIZADA
      P23: OPNAME
      P24: OPDATE;
```

```
ALUNOS_INSCRITOS_meta03 ::
      P1: COD_RAMO
      P2: COD_ESPECIALIZACAO
      P3: ANO
      P4: COD_REGIME_INGRESSO
      P5: E_TRABALHADOR_ESTUDANTE
      P6: E_MILITAR
      P7: E_DIRIGENTE_ASSOCIATIVO
      P8: E_DEPUTADO
      P9: E_ATLETA
      P10: E_EXTRAORDINARIO
      P11: E_BOLSEIRO
      P12: ANULADA
      P13: OPNAME
      P14: OPDATE
      P15: INSCRITO
      P16: CREDITO_PROPINAS;

CLASSIF_meta04 ::
      P1: ANOLEC
      P2: CLASS
      P3: CRED
      P4: TIPO
      P5: EQUIV
      P6: MELHORIA;

CLASSIF0_meta05 ::
      P1: ANOLEC
      P2: CLASS
      P3: CRED
      P4: TIPO
      P5: EQUIV
      P6: MELHORIA;

EQUIVALENCIAS_meta20 ::
      P1: CLASSIFICACAO
      P2: ANO_LECTIVO
      P3: TIPO_EQUIV
      P4: TIPO_DISC
      P5: CREDITOS
      P6: VERIFICADA;

ESTAGENG_meta23 ::
      P1: TEMA1
      P2: TEMA2
      P3: LOCAL1
      P4: LOCAL2
      P5: DATALIM;

INSCRICOES_meta27 ::
      P1: E_OPCAO
      P2: E_MELHORIA
      P3: TEM_SOBREPOSICAO
      P4: OPNAME
      P5: OPDATE
      P6: ANULADA;

INSCRICOES_EPOCA_meta29 ::
      P1: E_OPCAO
      P2: E_MELHORIA
      P3: E_ANTECIPADA;
```

```
NAO_INSCRITOS_meta32 ::
      P1: NOME
      P2: ANO;

VER_ALUNOS_meta46 ::
      P1: NOME
      P2: NOMEORD
      P3: ANO
      P4: DATANASC
      P5: BI
      P6: DATAVAC
      P7: NOMEPAI
      P8: NOMEMAE
      P9: RUA1
      P10: LOCAL1
      P11: TELEF1
      P12: RES1RES2
      P13: RUA2
      P14: LOCAL2
      P15: TELEF2
      P16: MNOME
      P17: MDATANASC
      P18: MNACIONAL
      P19: MBI
      P20: MLOCEMBI
      P21: MDATAVAC
      P22: MNOMEPAI
      P23: MNOMEMAE
      P24: MFREGNAT
      P25: MPAISNAT
      P26: MRUA1
      P27: MLOCAL1
      P28: MCODPOST1
      P29: MTELEF1
      P30: MFREG1
      P31: MRES1RES2
      P32: MRUA2
      P33: MLOCAL2
      P34: MCODPOST2
      P35: MTELEF2
      P36: MFREG2
      P37: VERIFICADO
      P38: OPDATE;

CURSOS_meta12 ::
      P1: NOME
      P2: GRAU
      P3: SIGLA
      P4: NUMANOS
      P5: ACTIVO
      P6: DIRTRAT
      P7: DIRNOME
      P8: DIREND1
      P9: DIREND2
      P10: DIREND3
      P11: TIPOMED
      P12: LIMITE
      P13: COD_POLO;

EQUIV0_meta19 ::
      P1: TIPO;
```

```
PAGINAS_CURSO_meta34 ::
      P1: PAGINA;

CPTERMOSDISC_meta11 ::
      P1: DATA
      P2: CRED
      P3: OPNAME
      P4: OPDATE
      P5: ESTADO
      P6: ALTERACAO
      P7: TIPO;

PAUTASDISC_meta37 ::
      P1: DATA1
      P2: DATA2
      P3: COMM
      P4: ESTADO
      P5: ALTERACAO
      P6: TIPO
      P7: NUM_CHAMADA;

TERMOSDISC_meta45 ::
      P1: DATA
      P2: CRED
      P3: OPDATE
      P4: ESTADO
      P5: ALTERACAO
      P6: TIPO
      P7: NUM_CHAMADA;

DISCS_meta14 ::
      P1: NOME
      P2: NOMEABR
      P3: TIPO_DISCIPLINA
      P4: FUNC_
      P5: ACTIVA
      P6: TERMOS
      P7: PAUTA_COLECTIVA;

EXAMES_meta24 ::
      P1: ANO
      P2: REGIME
      P3: DATA
      P4: HORA
      P5: SALA1
      P6: SALA2
      P7: SALA3
      P8: SALA4
      P9: SALA5
      P10: SALA6
      P11: SALA7;

PLANOS_meta41 ::
      P1: COMM
      P2: DATI
      P3: DATF
      P4: DURA
      P5: ANOI
      P6: COMM1
      P7: TIPO_PLANO;
```

```
PLANDISC_meta40 ::
      P1: AREA
      P2: CRED
      P3: FAC
      P4: ESCT
      P5: ESCTP
      P6: ESCSE
      P7: ESCP
      P8: REGIME
      P9: TIPO
      P10: COD_RAMO
      P11: COD_ESPECIALIZACAO;

OPERADORES_meta33 ::
      P1: NOME;

PERMISSOES_meta39 ::
      P1: COMENTARIO;

CONCELHOS_meta08 ::
      P1: NOME;

DEPARTS_meta13 ::
      P1: NOME
      P2: DIRTRAT
      P3: DIRNOME;

DOCS_meta16 ::
      P1: TRAT
      P2: NOME;

ESCOLAS_meta21 ::
      P1: NOME
      P2: PRESTRAT
      P3: PRESNOME;

ESTAB_meta22 ::
      P1: COD0
      P2: NOME
      P3: RUA
      P4: LOC
      P5: TELEF
      P6: FAX
      P7: TELEX
      P8: TRAT
      P9: NOMEABR
      P10: SIGLA;

FREGUESIAS_meta26 ::
      P1: NOME;

NUMERO_01 = seq of char;
CURSO_02 = seq of char;
DISC_09 = seq of char;
PLANO_08 = seq of char;
OPNAME_16 = seq of char;
CP1_06 = seq of char;
FREG1_07 = seq of char;
CPC_12 = seq of char;
DEPART_14 = seq of char;
CODIGO_DISTRITO_10 = seq of char;
SUPER_15 = seq of char;
```

```
ENDER_13 = seq of char;
COD = seq of char;
FREGN_03 = seq of char;
LOCDOCID_05 = seq of char;
PAISN_04 = seq of char;
TITULO_11 = seq of char;
ANO_LECTIVO = seq of char;
DATAHOM = seq of char;
ANOLEC = seq of char;
PAG = seq of char;
OFICIO = seq of char;
NUM_CHAMADA = seq of char;
FREGNAT = seq of char;
IDENT = seq of char;
EPOCA = seq of char;
CHAMADA = seq of char;
COD_FUNCAO = seq of char;
ANT_17 = seq of char;
ANO = seq of char;
PROGRAMA = seq of char;
LOCALIDADE = seq of char;
TRAT = seq of char;
NOME = seq of char;
END1 = seq of char;
END2 = seq of char;
END3 = seq of char;
DESCRICAO = seq of char;
LOCAL = seq of char;
ESTADO = seq of char;
QUANTOS = seq of char;
OPNAME = seq of char;
OPDATE = seq of char;
REGIME = seq of char;
CLASS = seq of char;
NUM_ORD = seq of char;
TIPOOF = seq of char;
QUEM = seq of char;
DATAENV = seq of char;
DATAREC = seq of char;
DATAPED = seq of char;
TIPO = seq of char;
SOBRE = seq of char;
MEDIA_PARCIAL = seq of char;
FACTOR_PONDERACAO = seq of char;
MATRICULA = seq of char;
INSCRICAO = seq of char;
SEGURO = seq of char;
CARTAO = seq of char;
GUIA = seq of char;
COTA = seq of char;
TOTAL = seq of char;
CLASSFINAL = seq of char;
CLASSESTAGIO = seq of char;
DATCONC = seq of char;
DATCARTA = seq of char;
RECCARTA = seq of char;
PROVISORIA = seq of char;
NOMEORD = seq of char;
SEXO = seq of char;
FALECIDO = seq of char;
DATN = seq of char;
E_PALOP = seq of char;
```

```
DOCID = seq of char;
NMPAI = seq of char;
NMMAE = seq of char;
RUA1 = seq of char;
LOC1 = seq of char;
TEL1 = seq of char;
RES1RES2 = seq of char;
RUA2 = seq of char;
LOC2 = seq of char;
TEL2 = seq of char;
DATM = seq of char;
COD_RAMO = seq of char;
COD_ESPECIALIZACAO = seq of char;
DATVACINA = seq of char;
ACTUALIZADA = seq of char;
COD_REGIME_INGRESSO = seq of char;
E_TRABALHADOR_ESTUDANTE = seq of char;
E_MILITAR = seq of char;
E_DIRIGENTE_ASSOCIATIVO = seq of char;
E_DEPUTADO = seq of char;
E_ATLETA = seq of char;
E_EXTRAORDINARIO = seq of char;
E_BOLSEIRO = seq of char;
ANULADA = seq of char;
INSCRITO = seq of char;
CREDITO_PROPINAS = seq of char;
CRED = seq of char;
EQUIV = seq of char;
MELHORIA = seq of char;
CLASSIFICACAO = seq of char;
TIPO_EQUIV = seq of char;
TIPO_DISC = seq of char;
CREDITOS = seq of char;
VERIFICADA = seq of char;
TEMA1 = seq of char;
TEMA2 = seq of char;
LOCAL1 = seq of char;
LOCAL2 = seq of char;
DATALIM = seq of char;
E_OPCAO = seq of char;
E_MELHORIA = seq of char;
TEM_SOBREPOSICAO = seq of char;
E_ANTECIPADA = seq of char;
DATANASC = seq of char;
BI = seq of char;
DATAVAC = seq of char;
NOMEPAI = seq of char;
NOMEMAE = seq of char;
TELEF1 = seq of char;
TELEF2 = seq of char;
MNOME = seq of char;
MDATANASC = seq of char;
MNACIONAL = seq of char;
MBI = seq of char;
MLOCEMBI = seq of char;
MDATAVAC = seq of char;
MNOMEPAI = seq of char;
MNOMEMAE = seq of char;
MFREGNAT = seq of char;
MPAISNAT = seq of char;
MRUA1 = seq of char;
MLOCAL1 = seq of char;
```

```
MCODPOST1 = seq of char;
MTELEF1 = seq of char;
MFREG1 = seq of char;
MRES1RES2 = seq of char;
MRUA2 = seq of char;
MLOCAL2 = seq of char;
MCODPOST2 = seq of char;
MTELEF2 = seq of char;
MFREG2 = seq of char;
VERIFICADO = seq of char;
GRAU = seq of char;
SIGLA = seq of char;
NUMANOS = seq of char;
ACTIVO = seq of char;
DIRTRAT = seq of char;
DIRNOME = seq of char;
DIREND1 = seq of char;
DIREND2 = seq of char;
DIREND3 = seq of char;
TIPOMED = seq of char;
LIMITE = seq of char;
COD_POLO = seq of char;
PAGINA = seq of char;
DATA = seq of char;
ALTERACAO = seq of char;
DATA1 = seq of char;
DATA2 = seq of char;
COMM = seq of char;
NOMEABR = seq of char;
TIPO_DISCIPLINA = seq of char;
FUNC_ = seq of char;
ACTIVA = seq of char;
TERMOS = seq of char;
PAUTA_COLECTIVA = seq of char;
HORA = seq of char;
SALA1 = seq of char;
SALA2 = seq of char;
SALA3 = seq of char;
SALA4 = seq of char;
SALA5 = seq of char;
SALA6 = seq of char;
SALA7 = seq of char;
DATI = seq of char;
DATF = seq of char;
DURA = seq of char;
ANOI = seq of char;
COMM1 = seq of char;
TIPO_PLANO = seq of char;
AREA = seq of char;
FAC = seq of char;
ESCT = seq of char;
ESCTP = seq of char;
ESCSE = seq of char;
ESCP = seq of char;
COMENTARIO = seq of char;
PRESTRAT = seq of char;
PRESNOME = seq of char;
COD0 = seq of char;
RUA = seq of char;
LOC = seq of char;
TELEF = seq of char;
FAX = seq of char;
```

```
TELEX = seq of char;

state STATE of
   x : Sets
init s ==
   s = mk_STATE(mk_Sets({|->},{|->},{|->},{|->},{|->},{|->},{|->},{|->},{|->},{|->},{|->},{|->},{|

end

functions
```